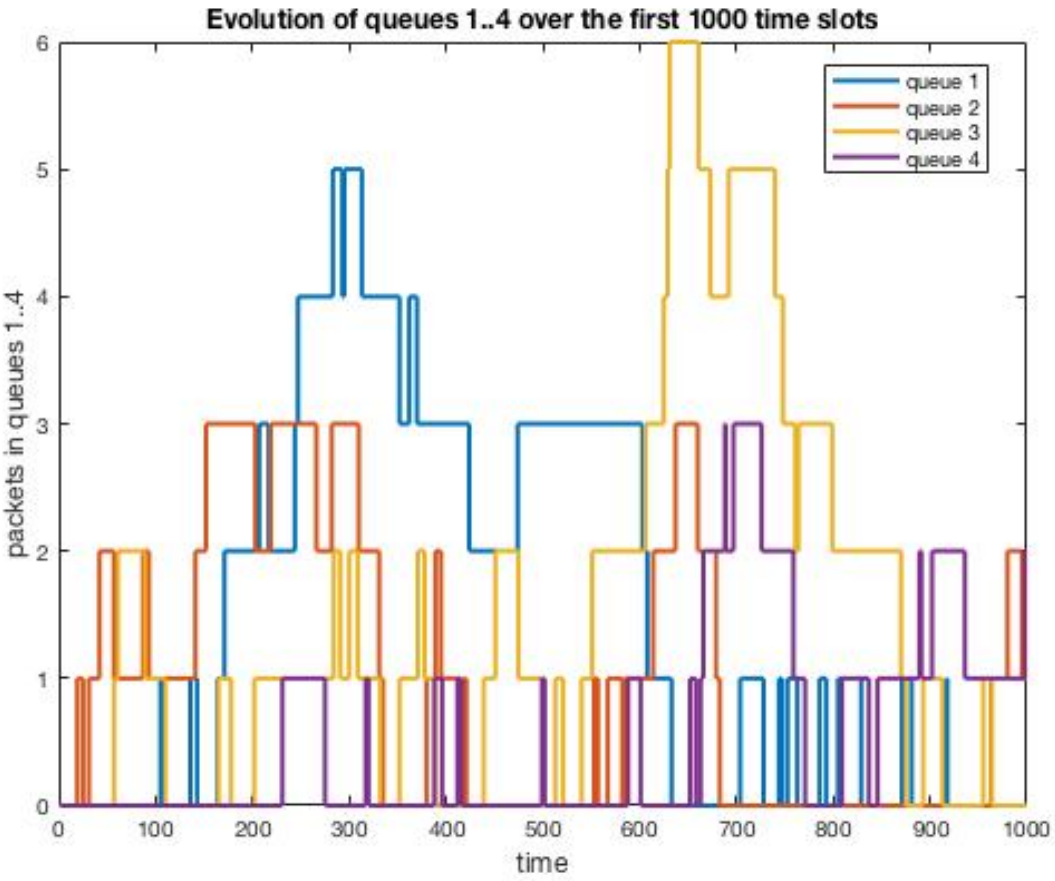
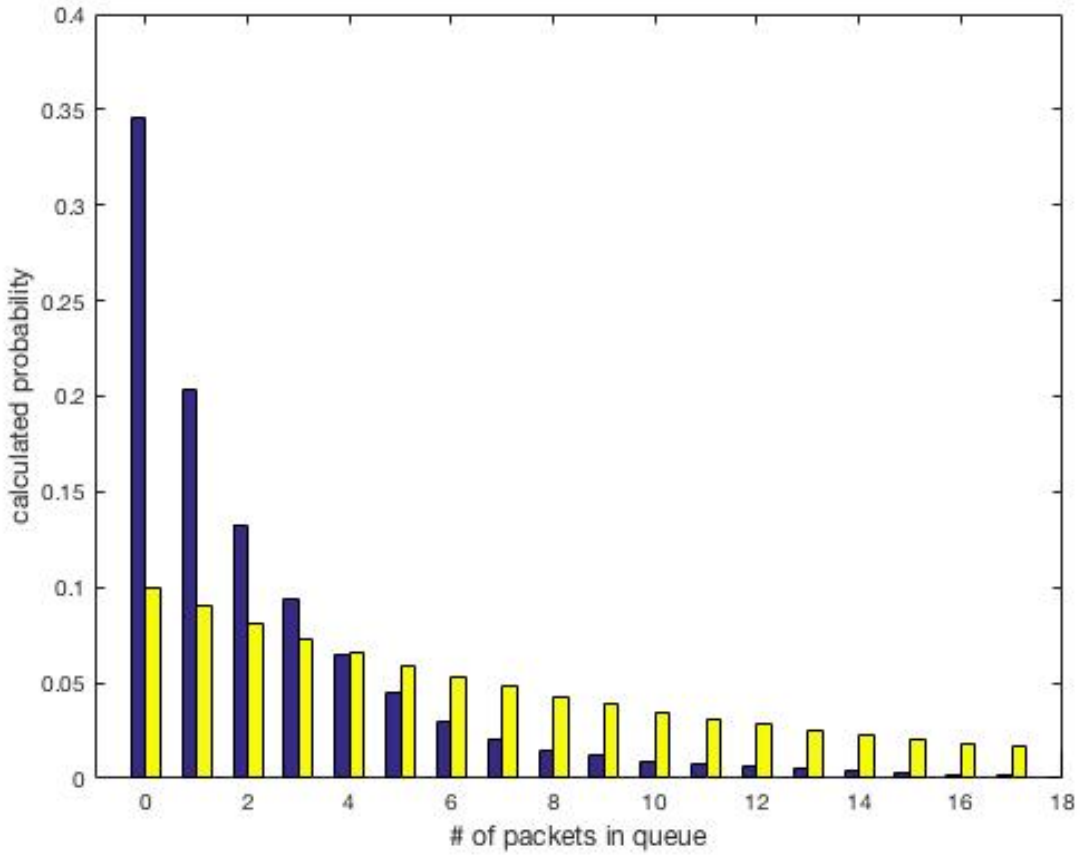
G)



The plot is shown above.

H)



From the plot we see that the dominant assumption underestimates the performance of RA policy.
Also without the assumption, the empty queue remains silent this further reduce the burden of the system. So in reality RA should perform better.

Code:

```
function hw5
%     question1
%     question2
%     question5
%     question7G()
    question7H()


end


function question1
    pM=[(1/4) (3/4) ;1/5 4/5];
    pM^4
    pM^5
%     converge!
    [0.2105 0.7895]*pM
end

function quesiton2
    p=[0 1/2 1/2; 1/4 1/2 1/4; 1/4 1/4 1/2];
    p^7
    p^8
%     p^4
    [0.2 0.4 0.4]*p
end

function question5
    p=[1/3 1/6 1/2; 0 1 0; 0 0 1];
    p^100
%     [1,0,0]*p^100
%     [0,0,1]*p
    [0,1,0]*p
%     [0,0.5,0.5]*p
end



function [x] = aloha_uplink_simulation(J,p,lambda,N)
    % Implementing no concurrence hypothesis is very difficult.
    % Instead, we assume arrivals have precedence over service.
    x=zeros(J,N);
    for t=1:N-1
        arrivals=binornd(1,lambda,J,1);
        is_there_packets=x(:,t)>0;
        decide_to_transmit=binornd(1,p,J,1);
        service = is_there_packets & decide_to_transmit &   ~arrivals;
        % service = is_there_packets & decide_to_transmit;
        % Uncomment the line above to relax the non-concurrence assump.
        if sum(service)<=1
            x(:,t+1)=x(:,t)+arrivals-service;
        else
            x(:,t+1)=x(:,t)+arrivals;
        end
    end
end


function question7G
clear all; close all; clc;
 J=16;
```

```
 p=1/J;
 N=10^5;
 lambda=0.9*p*(1-p)^(J-1);
 x = aloha_uplink_simulation(J,p,lambda,N);
 figure
 stairs(1:1000,x(1:4,1:1000)','LineWidth',2);
 title('Evolution of queues 1..4 over the first 1000 time slots','FontSize',12)
 xlabel('time','FontSize',12)
 ylabel('packets in queues 1..4','FontSize',12)
 legend('queue 1','queue 2','queue 3','queue 4','Location','Best')

end

function question7H
    J=16;
    p=1/J;
    N=10^5;
    lambda=0.9*p*(1-p)^(J-1);
    x = aloha_uplink_simulation(J,p,lambda,N);
    Q1=max(x(1,:));
    frequencies=zeros(1,Q1+1);
    for i=0:Q1
        frequencies(1,i+1)=sum(x(1,:)==i);
    end
    rho=lambda/(p*(1-p)^(J-1));
    A=[frequencies/N;(1-rho)*(rho.^(0:Q1))];
    A=A';
    figure
    bar(0:Q1,A, 1);
    axis([-1 Q1 0 0.4])
    xlabel('# of packets in queue','FontSize',12)
    ylabel('calculated probability','FontSize',12)
end
```